REMARKS/ARGUMENTS

Reconsideration of this application in view of the above amendments and the remarks below is respectfully requested. By this amendment, Claims 1, 16, 18 and 21 have been amended. No claims have been added or cancelled. Hence, Claims 1-10 and 15-21 are pending in the application.

SUMMARY OF THE REJECTIONS/OBJECTIONS

Claim 1 is rejected under 35 U.S.C. 112, first paragraph, as allegedly failing to comply with the written description requirement.

Claims 1-5 and 8-10 are rejected under 35 U.S.C. 102(b) as allegedly anticipated by Burroughs et al, U.S. Patent No. 6,076,090 (hereinafter *Burroughs*).

THE REJECTIONS NOT BASED ON THE PRIOR ART

Claim 1 is rejected under 35 U.S.C. 112, first paragraph, as allegedly failing to comply with the written description requirement. Specifically, the Office Action asserts that "[w]hile claim 1 recites the limitation of 'determining that collection of data elements to be returned by the query corresponds to a first data structure not defined with[in] a type dictionary of the database system,' the Office is unable to make a determination as to the portion of the specification which would reasonably convey to one o[f] ordinary skill in the art said limitation."

Claim 1 has been amended to read, among other things, ".... containing data fields, wherein the data fields are not specified by a data type definition within a type dictionary of the database system...". Applicant submits that, in view of Applicant's specification in paragraphs 8 (page 6 lines 1-3 and 14-16), 11 (page 9 lines 12-25), and 23 (page 19 lines 6-12), present

Claim 1 satisfies the written description requirement under 35 U.S.C. 112, first paragraph. Accordingly, removal of the rejection is respectfully requested.

THE REJECTIONS BASED ON THE PRIOR ART

102(b) - *Burroughs*

Claims 1-5 and 8-10 are rejected under 35 U.S.C. 102(b) as allegedly anticipated by Burroughs et al, U.S. Patent No. 6,076,090 (hereinafter *Burroughs*). Applicant submits that Claims 1-5 and 8-10, as amended, are patentable over *Burroughs*. Each of the pending claims is discussed herein.

Claim 1

Independent method claim 1 recites:

receiving a request to execute a query;
determining that a collection of data elements to be returned by the query corresponds to a first data structure **containing data fields, wherein the data fields are not specified by a data type definition** within a type dictionary of the database system;
obtaining attribute values that respectively describe the data fields within the first data structure; and
recording, within the type dictionary, a first data type definition that specifies the data fields described by the attribute values (emphasis added).

The method of Claim 1 provides an advantageous way for a database system to dynamically obtain and record data type information (or attribute values) that specifies data fields in data elements to be returned by a query, even if the data fields are not specified by a data type definition within the database system prior to receiving a request for the query. According to Claim 1, data elements to be returned by a requested query may be determined. The determined data elements may correspond to a data structure that contains data fields not specified by a data type definition within a type dictionary of the database system. Hence, attribute values that respectively describe the data fields (not specified in the type dictionary)

may be obtained. Accordingly, a data type definition by which the data fields are specified may then be recorded within the type dictionary of the database system.

Thus, under one embodiment of the present invention, if the data elements to be returned by the requested query are, for example, of an opaque data type such as a binary large object (blob) that contains data fields not specified by any data type definition within a type dictionary of a database system, a data type definition that specifies the data fields in the opaque data type may still be recorded in the type dictionary. As a result, the query result to be returned to a requestor may contain a better description about the data fields contained in the opaque object.

Such a method is neither disclosed nor suggested by *Burroughs*. Instead, *Burroughs* discloses a method of translating Java language types into database types and then storing the translated Java language types in corresponding columns of a relational table to persist an object (col. 3 lines 62-64).

As disclosed by *Burroughs*, fields within the object are examined using Java Reflection methods (col. 3 lines 57-61). The primitive language fields (such as integer, character, long, float, string, etc.) found in the object are stored in columns of corresponding types in a relational table (FIG. 8 and its accompanying text). Similarly, a complex field in the object, such as Hashtable and Vector types in Java, may be stored in a column of binary format in the form of a (bit) stream (col. 4 lines 20-23).

According to *Burroughs*, a row in the relational table previously mentioned may be retrieved in a restoring transaction, to recover the object (FIG. 9 and col. 14 lines 39-55). Notably, in the retrieval (i.e., what *Burroughs* refers to as a restoring transaction), only an object identifier is needed to retrieve the row information stored in the relational table. Unlike the storing transaction, retrieval (or a restoring transaction) does not involve retrieving attribute

values for the fields of the Java object using the Java Reflection methods, but only involves retrieving row data from the relational table.

Similarly, according to *Burroughs*, the persisted object may be deleted (FIG. 10 and col. 14, line 56-col. 15, line 2) in a delete transaction. Like the retrieval, to accomplish the deletion, an object identifier is used to identify the row of the relational table that corresponds to the object, and the row in the relational table is simply deleted. Again, no translation by the Java Reflection methods is needed to support deleting the object.

Type Dictionary

The method disclosed by *Burroughs* as discussed above is quite different from that claimed in amended Claim 1. For example, Claim 1 features a type dictionary that stores data type definitions. On the other hand, there is no disclosure in *Burroughs* about a type dictionary.

More specifically, Claim 1 features a type dictionary that might not initially have a data type definition that specifies data fields of a data structure, but is then recorded with a data type definition relating to the same data fields. *Burroughs*, on the other hand, simply fails to disclose a type dictionary of a database system, let alone a type dictionary that dynamically accepts data type definitions, previously non-existent, for data fields of data structures.

Data Types Recorded In Response to Retrieving Data Elements of Unknown Types

Furthermore, data fields of Claim 1 are initially not specified within the type dictionary of the database system, but later are specified by a data definition that is recorded in the type dictionary.

On the other hand, the columns of the relational table in *Burroughs*, which are analogous to data elements according to the Office Action, use only pre-existing data types. As disclosed by *Burroughs,* neither data types that correspond to primitive Java language types nor the bit format data type that corresponds to Hashtable and Vector in Java are previously

unspecified and later specified within a type dictionary. In fact, *Burroughs* only discloses 1) a type of columns corresponding to primitive Java language types and 2) a type of columns (i.e., streams in binary format) corresponding to complex Java language types. However, both types of columns are defined and supported in the database system *before* and *after* the object is to be retrieved.

Obtaining Attributes

Claim 1 features obtaining attributes that describe data fields *to be returned by the requested query*, i.e., at the time of retrieval.

However, the translation in *Burroughs* occurs when the object is being stored into the relational table, not when the object is retrieved from its persistent store. Thus, when a retrieval of the object is performed in *Burroughs*, the translation has already been performed, and there could not have been any unspecified data types, since all the unknown data types, if there were such, would have been resolved by the translation using Java Reflection methods in an earlier storing transaction.

Thus, at most, *Burroughs* could only disclose obtaining attributes about fields of unknown data types at the time of inserting row data in the relational table, not at the time of retrieval. That is not the same as what is claimed by Claim 1.

Assertions in the Office Action

Contrary to the contention at paragraph 9 (page 3) of the Office Action that col. 3 lines 37-44 and 53-65 of *Burroughs* discloses the above discussed features of Claim 1, Applicant respectfully submits that the cited passage of *Burroughs* only pertains to the **storing** transaction (i.e., a transaction that persists an object), and is not related to the **restoring** transaction (i.e., retrieval). Thus, the cited passages cannot possibly be about a method practiced in response to receiving a query as featured in Claim 1.

Overall, it should come as no surprise that Claim 1 is not disclosed or suggested by *Burroughs*. *Burroughs* addresses a very different problem. *Burroughs* is concerned with persisting an object in a relational table using existing data types in a database system such as those corresponding to primitive Java language types, or Hashtable and Vector. On the other hand, the method of Claim 1 is concerned with how to determine unspecified data fields in an opaque object, obtain attributes values for the data fields, and record a new data type definition that specifies the previously unspecified data fields using a type dictionary. Since *Burroughs* addresses a very different problem, it is no surprise that *Burroughs* provides a very different solution. For the reasons given above, Applicant submits that Claim 1, as amended, is patentable over *Burroughs*.

Claims 18 and 21

Claims 18 and 21 are database system and computer-readable medium claims, which are analogous to the method Claim 1. Applicant submits that Claims 18 and 21 are patentable over *Burroughs* for at least the same reasons as those given above in connection with Claim 1.

Claims 2-5, 8-10, 19 and 20

Claims 2-5, 8-10, 19, and 20 depend from, and hence, incorporate all of the limitations of Claim 1 or 18. These claims also recite further limitations that render them patentable over *Burroughs*. Applicant submits that these claims are patentable over *Burroughs* for at least the reasons given above in connection with Claim 1.

103(a) – *Burroughs* in View of Official Notice

Claims 6, 7 and 15-17 are rejected under 35 U.S.C. 103(a) as allegedly unpatentable over *Burroughs*, in view of Official Notice. The rejection is respectfully traversed.

Claims 6 and 7

The Office Action takes Official Notice as follows:

> [I]t would have been obvious to one of ordinary skill in the art at the time the invention was claimed that a binary large object (i.e., "blob") be returned, wherein a blob is a collection of binary data stored as a single entity in a database management system. Therefore, where it would have been obvious to one of ordinary skill in the art at the time the invention was claimed that an attribute value describe a blob, it would have been obvious to one of ordinary skill in the art that said attribute value(s) be returned accordingly.

However, Claim 6 recites

> The method of claim 1 further comprising determining whether any of the attribute values describes a data field having a plurality of component data fields.

Thus, even if the Official Notice were correct, the Notice still fails to disclose a step of determining whether any of the attribute values describes a data field *having a plurality of component data*, as recited in Claim 6.

Likewise, Claim 7 recites

> The method of claim 6 further comprising obtaining attribute values that describe the plurality of component data fields.

Thus, even if the Official Notice were correct, the Notice still fails to disclose a an obtaining step for the attribute values describes a data field *having a plurality of component data*, as recited in Claim 7.

Applicant respectfully submits that the Official Notice has been improperly taken. According to MPEP, "Official notice unsupported by documentary evidence should only be taken by the examiner where the facts asserted to be well-known, or to be common knowledge in the art are capable of instant and unquestionable demonstration as being well-known" (2144.03.A). Applicant respectfully requests that support be provided as to why the alleged fact taken under the Official Notice is well known in the art or capable of instant and unquestionable demonstration.

Furthermore, Claims 6 and 7 depend from, and hence, incorporate all of the limitations of Claim 1. For the reasons set forth above, Applicant respectfully submits that Claims 6 and 7 are patentable over *Burroughs* in view of the Official Notice.

Claim 15

The Official Action also takes Official Notice as follows:

> [I]t would have been obvious to one of ordinary skill in the art at the time the invention was claimed that when a function such as a SQL statement is executed, a collection of aggregate data values is returned.

However, Claim 15 recites

> The method of claim 1 wherein receiving a request to execute a query comprises receiving a request to execute a function that returns a collection of aggregate data values.

Thus, even if the Official Notice were correct, the Notice still fails to disclose receiving a request to execute a function that returns a collection of aggregate data values, as recited in Claim 15.

Applicant respectfully submits that the Official Notice has been improperly taken. According to MPEP, "Official notice unsupported by documentary evidence should only be taken by the examiner where the facts asserted to be well-known, or to be common knowledge in the art are capable of instant and unquestionable demonstration as being well-known" (2144.03.A). Applicant respectfully requests that support be provided as to why the alleged fact taken under the Official Notice is well known in the art or capable of instant and unquestionable demonstration.

Furthermore, Claim 15 depends from, and hence, incorporates all of the limitations of Claim 1. For the reasons set forth above, Applicant respectfully submits that Claim 15 is patentable over *Burroughs* in view of the Official Notice.

Claim 16

The Official Action further takes Official Notice as follows:

> [I]t would have been obvious to one of ordinary skill in the art at the time the invention was claimed that queries commonly indicate the type of value (e.g., an integer, text, or string) to be returned by a query.

However, Claim 16 recites

> The method of claim 1 wherein determining that a collection of data elements to be returned by the query corresponds to a first data structure containing data fields not defined within a type dictionary of the database system comprises determining that a predetermined return type is indicated by the query.

Thus, even if the Official Notice were correct, the Notice still fails to disclose determining that a predetermined return type is indicated by the query. In fact, as commonly known, results from a query do not necessarily indicate any type information, for example, where the results are displayed on a monitor in ASCII.

Applicant respectfully submits that the Official Notice has been improperly taken. According to MPEP, "Official notice unsupported by documentary evidence should only be taken by the examiner where the facts asserted to be well-known, or to be common knowledge in the art are capable of instant and unquestionable demonstration as being well-known" (2144.03.A). Applicant respectfully requests that support be provided as to why the alleged fact taken under the Official Notice is well known in the art or capable of instant and unquestionable demonstration.

Furthermore, Claim 16 depends from, and hence, incorporates all of the limitations of Claim 1. For the reasons set forth above, Applicant respectfully submits that Claim 16 is patentable over *Burroughs* in view of the Official Notice.

Claim 17

Claim 17 depends from, and hence, incorporates all of the limitations of Claim 1. Claim 17 also recites further limitations that render it patentable over *Burroughs*. Applicant submits that Claim 17 patentable over *Burroughs* in view of the Official Notice for at least the reasons given above in connection with Claim 1.

For the reasons set forth above, it is respectfully submitted that all of the pending claims are now in condition for allowance. Therefore, the issuance of a formal Notice of Allowance is believed next in order, and that action is most earnestly solicited.

The Examiner is respectfully requested to contact the undersigned by telephone if it is believed that such contact would further the examination of the present application.

Please charge any shortages or credit any overages to Deposit Account No. 50-1302.

Respectfully submitted,
HICKMAN PALERMO TRUONG & BECKER LLP

Zhichong Gu
Reg. No. 56,543

2055 Gateway Place, Suite 550
San Jose, CA 95110
(408) 414-1080
Date: April 2, 2007
Facsimile: (408) 414-1076

CERTIFICATE OF MAILING
I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Mail Stop Amendment, Commissioner for Patents, P. O. Box 1450, Alexandria, VA 22313-1450
on 4/2/07 by Martina Placid
Martina Placid